Proceeding of International Conference on Electrical Engineering, Computer Science and Informatics (EECSI 2015), Palembang, Indonesia, 19 -20 August 2015

NUMERICAL SOLUTION FOR SOLVING SPACE-FRACTIONAL DIFFUSION EQUATIONS USING HALF-SWEEP GAUSS-SEIDEL ITERATIVE METHOD

A. Sunarto
Faculty of Science and Natural Resources,
Universiti Malaysia Sabah,
88400 Kota Kinabalu, Sabah, Malaysia
andang99@gmail.com

J. Sulaiman
Faculty of Science and Natural Resources,
Universiti Malaysia Sabah,
88400 Kota Kinabalu,Sabah, Malaysia
jumat@ums.edu.my

A. Saudi
Faculty of Computing and Informatics,
Universiti Malaysia Sabah,
88400 Kota Kinabalu,Sabah, Malaysia
azali60@gmail.com

***Abstract*—The main purpose of this paper is to examine the effectiveness of Half-Sweep Gauss-Seidel (HSGS) method for Space-Fractional diffusion equations. The Caputo's derivative and implicit finite difference scheme will be used to discretize linear space-fractional equation of the first order to construct system linear equation. The basic formulation and application of the HSGS iterative method are also presented. Two numerical examples and comparison with other iterative methods shows that the present method is effective. Based on computational numerical result, the solution obtained by proposed iterative method is in excellent agreement, it can be concluded that the proposed iterative method is superior to the Full-Sweep Gauss-Seidel (FSGS) iterative method**

Keywords—Caputo's fractional derivative; Implicit finite difference Schemee; HSGS method

INTRODUCTION

Generally, the first order linear space-fractional partial diffusion equations (SFPDE's) can be defined as follows

$$\frac{\partial U(x,t)}{\partial t} = a(x)\frac{\partial^{\beta} U(x,t)}{\partial x^{\beta}} + b(x)\frac{\partial U(x,t)}{\partial x} + c(x)U(x,t) + f(x,t) \qquad (1)$$

with initial condition

$$U(x,0) = f(x),\ 0 \le x \le \lambda,$$

and boundary conditions

$$U(0,t) = g_0(t),\ U(\lambda,t) = g_1(t),\ 0 < t \le T.$$

Many natural phenomena in physic, engineering and other sciences can be presented very successfully by models using fractional calculus [1,2,3]. Generally, fractional partial differential equations can be arranged into two important types: time-fractional partial differential equations (TFPDE's) and space-fractional partial differential equations (SFPDE's). Solution of space-fractional partial diffusion equation (SFPDE's) have been studied by many authors. For instance, Azizi and Loghmani [4] used Chebyshev collocation method to discretize space-fractional to obtain a linear system of ordinary differential equation and used the finite difference for solving the resulting system. Besides that, methods such as finite difference Theta method [5], Tau approach [6], explicit finite difference scheme [7], are also analysed in solving SFPDE's. Nevertheless, these methods lead to dense linear system and can be too expensive to solve large linear system. As a result, iterative methods have been proposed to get numerical solution of the linear system. However, these iterative methods are based on the standard or Full-Sweep iterative methods which are more expensive in term of computation time. For that reason, in this paper a discretization scheme namely Half-Sweep approach is applied to discretize Eq.(1) to generate of linear system.

Before constructing the linear systems, some definitions that can be applied for fractional derivative theory need to developing the approximation equation of Problem (1).

Definition 1.[8] The Riemann-Liouville fractional integral operator , J^{β} of order-β is defined as

$$J^{\beta} f(x) = \frac{1}{\Gamma(\beta)} \int_0^x (x-t)^{\beta-1} f(t)dt,\ \beta > 0\ , x > 0 \qquad (2)$$

Definition 2.[4,8] The Caputo's fractional partial derivative operator, D^{β} of order -β is defined as

$$D^{\beta} f(x) = \frac{1}{\Gamma(m-\beta)} \int_0^x \frac{f^{(m)}(t)}{(x-t)^{\beta-m+1}} dt, \ \beta > 0 \tag{3}$$

with $m-1<\beta\le m, m\in N, \ x>0$.

We have the following properties when $m-1<\beta\le m$, $x>0$:

$D^{\beta} k = 0$, (k is constant),

$$D^{\beta} x^n = \begin{cases} 0, & \text{for } n\in N_0 \text{ and } n<[\beta] \\ \frac{\Gamma(n+1)}{\Gamma(n+1-\beta)} x^{n-\beta}, & \text{for } n\in N_0 \text{ and } n\ge[\beta] \end{cases}$$

where function $[\beta]$ denotes the smallest integer greater than or equal to β, N0= $\{0,1,2,...\}$ and $\Gamma(.)$ is the gamma function.

A part from the standard iterative methods as mentioned in the paragraph, the proposed HSGS iterative method is inspired by concept of Half-Sweep iteration which was introduced by Abdullah [9] via the Explicit Decoupled Group (EDG) iterative method to solve two-dimensional Poisson equations. The application of Half-Sweep iterative methods have been implemented by Sulaiman *et al* [10], Aruchunan and Sulaiman [11], Muthuvalu and Sulaiman [12]. In this paper, we examine the applications of Half-Sweep Gauss-Seidel (HSGS) iterative method to solve space-fractional partial differential equations (SFPDE's) based on the Caputo's implicit finite difference approximation equation. To show the performance of the Half-Sweep Gauss-Seidel method, we also implement the Full-Sweep Gauss-Seidel (FSGS) iterative methods being used as a control method.

CAPUTO'S IMPLICIT FINITE DIFFERENCE APPROXIMATION

Consider $h=\frac{\lambda}{k}$, where k is positive integer number. By using second order difference approximation equations, we get

$$\frac{\partial^{\beta} U(x_i,t_n)}{\partial x^{\beta}} = \frac{1}{\Gamma(2-\beta)} \int_0^{t_n} \frac{\partial^2 U(x_i,s)}{\partial x^2} (t_n-s)^{1-\beta} \partial s$$

$$= \frac{1}{\Gamma(2-\beta)} \sum_{j=0,2,4}^{i-2} \int_{jh}^{(j+1)h} \left(\frac{U_{i-j+2,n}-2U_{i-j,n}+U_{i-j-2,n}}{2h^2} \right) (nh-s)^{\beta} \partial s$$

$$= \frac{(2h)^{-\beta}}{\Gamma(3-\beta)} \sum_{j=0,2,4}^{i-2} \left(U_{i-j+2,n}-2U_{i-j,n}+U_{i-j-2,n}\right) \left(\left(\frac{j}{2}+1\right)^{2-\beta} - \frac{j}{2}^{2-\beta} \right) \tag{4}$$

where $\sigma_{\beta,2h} = \frac{(2h)^{-\beta}}{\Gamma(3-\beta)}$ and $g_j^{\beta} = \left(\frac{j}{2}+1\right)^{2-\beta} - \frac{j}{2}^{2-\beta}$.

Subsequently we get the discrete approximation of Eq.(4) being given as

$$\frac{\partial^{\beta} U(x_i,t_n)}{\partial x^{\beta}} = \sigma_{\beta,2h} \sum_{j=0,2,4}^{i-2} g_j^{\beta} \left(U_{i-j+2,n}-2U_{i-j,n}+U_{i-j-2,n}\right)$$

Using Caputo's implicit finite difference approximation, we approximate Problem (1) as

$$\lambda\left(U_{i,n}-U_{i,n-2}\right) = a_i \sigma_{\beta,2h} \sum_{j=0.2.4}^{i-2} g_j^{\beta} \left(U_{i-j+2,n}-2U_{i-j,n}+U_{i-j-2,n}\right) + b_i \frac{\left(U_{i+2,n}-U_{i-2,n}\right)}{4h} + C_i U_{i,n} + f_{i,n}$$

for *i* = *2,4,...m-2*. Thus, let us simplify the above approximation equation as

$$\lambda U_{i,n-2} = -a_i \sigma_{\beta,2h} \sum_{j=0,2,4}^{i-2} g_j^{\beta} \left(U_{i-j+2,n}-2U_{i-j,n}+U_{i-j-2,n}\right) - \frac{b_i}{4h}\left(U_{i+2,n}-U_{i-2,n}\right) - C_i U_{i,n} + \lambda U_{i,n} - f_{i,n} \tag{5}$$

From Eq.(5), it can be rewritten in simple form as

$$\therefore b_i^* U_{i-2,n} + \left(\lambda - c_i^*\right) U_{i,n} - b_i^* U_{i+2,n_i} - a_i^* \sum_{j=0,2,4}^{i-2} g_j^{\beta} \left(U_{i-j+2,n}-2U_{i-j,n}+U_{i-j-2,n}\right) = f_i \tag{6}$$

where $a_i^* = a_i \sigma_{\beta,2h}$, $b_i^* = \frac{b_i}{4h}$, $c_i^* = c_i$, $F_i^* = f_{i,n}$ and $f_i = \lambda\left(U_{i,n-2}\right) + F_i^*$

Based on Eq.(6), the approximation equation is known as the fully half-sweep implicit finite difference approximation equation which is consistent and second order accuracy in space-fractional. Let us define Eq.(6) for $n>3$ being rewritten as

$$-R_i + \alpha_i U_{i-6,n} + s_i U_{i-4} + p_i U_{i-2,n} + q_i U_{i,n} + r_i U_{i+2,n} = f_i \tag{7}$$

where

$$R_i = a_i^* \sum_{j=6}^{i-2} g_j^{\beta} \left(U_{i-j+2} - 2U_{i-j,n} + U_{i-j-2,n}\right),$$

$$\alpha_i = \left(-a_i^* g_2^{\beta}\right),$$

$$s_i = \left(-a_i^* g_1^{\beta} + 2a_i^* g_2^{\beta}\right),$$

$$p_i = \left(b_i^* - a_i^* g_2^{\beta} + 2a_i^* g_1^{\beta} - a_i^*\right),$$

$$q_i = \left(-a_i^* g_1^{\beta} + 2a_i^* + \left(\lambda - c_i^*\right)\right),$$

$$r_i = \left(-a_i^* - b_i^*\right)$$

By considering Eq.(7) over interior point in solution domain Eq.(1) , we can construct a linear system in matrix form as

$$A \underset{\sim}{U} = \underset{\sim}{f} \tag{8}$$

where
$$A = \begin{bmatrix} q_2 & r_2 & & & & & & & \\ p_4 & q_4 & r_4 & & & & & & \\ s_6 & p_6 & q_6 & r_6 & & & & & \\ \alpha_8 & s_8 & p_8 & q_8 & r_8 & & & & \\ & \alpha_{10} & s_{10} & p_{10} & q_{10} & r_{10} & & & \\ & & 0 & 0 & 0 & 0 & 0 & & \\ & & & \alpha_{m-4} & s_{m-4} & p_{m-4} & q_{m-4} & r_{m-4} \\ & & & & \alpha_{m-2} & s_{m-2} & p_{m-2} & q_{m-2} \end{bmatrix}_{(m-2)x(m-2)}$$

$$\underset{\sim}{U} = \begin{bmatrix} U_{2,1} & U_{4,1} & U_{6,1} & \Lambda & U_{m-4,1} & U_{m-2,1} \end{bmatrix}^T,$$

$$\underset{\sim}{f} = \begin{bmatrix} f_2 - p_2 U_{2,1} & f_4 + s_4 U_{41} & f_6 + \alpha_6 U_{6,1} & f_8 + R_i & \Lambda & f_{m-4,1} + R_{m-4} & f_{m-2,1} - p_{m-2} U_{m,1} + R_{m-2} \end{bmatrix}^T$$

HALF-SWEEP GAUSS-SEIDEL ITERATIVE METHOD

As mentioned above, the generated linear system of Eq.8 will be solved by using Half-Sweep Gauss-Seidel (HSGS) iterative method. Let the coefficient matrix, A, be decomposed into

$$A = D - L - V \tag{9}$$

where D, L and V are diagonal, strictly lower triangular and strictly upper triangular matrices respectively [13,14,15]. Therefore, the general scheme for the HSGS iterative method can be written as [10,13,14,15,16]

$$\underset{\sim}{U}^{(k+1)} = (D-L)^{-1}\left(V \underset{\sim}{U}^{(k)} + \underset{\sim}{f}\right) \tag{10}$$

where $\underset{\sim}{U}^{(k)}$ represent unknown vector at k[th] iteration.

By determining value of matrices D, L and V as states in Eq.8, the proposed algorithm for Half-Sweep Gauss-Seidel iterative method to solve Eq.8 generally can be described in Algorithm 1

Algorithm 1 : Half-Sweep Gauss-Seidel Algorithm

i. Initialize $\underset{\sim}{U} \leftarrow 0$ and $\varepsilon \leftarrow 10^{-10}$
ii. For i= 2,4,…,m-2 and j= 0,2,…,n-1 assign
$$\underset{\sim}{U}^{(k+1)} = (D-L)^{-1}\left(V \underset{\sim}{U}^{(k)} + \underset{\sim}{f}\right)$$
iii. Convergence test. If the convergence criterion i.e $\left\| \underset{\sim}{U}^{(k+1)} - \underset{\sim}{U}^{(k)} \right\| \le \varepsilon = 10^{-10}$ is satisfied, go step (iv). Otherwise go back to step (ii)
iv. Display approximate solutions

NUMERICAL EXPERIMENTS

In order to evaluate the effectiveness of the proposed method, two examples were carried out. Three parameters such as number of iterations (K), execution time (second) and maximum error at three different values of $\beta = 1.2$, $\beta = 1.5$ and $\beta = 1.7$ are considered as measurement to evaluate the performance of the proposed methods. The Full-Sweep Gauss-Seidel (FSGS) methods was used as the control of comparison of numerical results. Throughout the numerical simulation, the convergence test was carried out with tolerance of $\varepsilon = 10^{-10}$ with several mesh sizes as 128, 256, 512, 1024 and 2048.

Example 1 : [4]

Let us consider the following space-fractional initial boundary value problem

$$\frac{\partial U(x,t)}{\partial t} = d(x)\frac{\partial^\beta U(x,t)}{\partial x^\beta} + p(x,t), \tag{11}$$

On finite domain $0 < x < 1$, with the diffusion coefficient $d(x) = \Gamma(\beta) x^{0.5}$.

The source function $p(x,t) = (x^2+1)\cos(t+1) - 2x\sin(t+1)$, with the initial condition $U(x,0) = (x^2+1)\sin(1)$ and the boundary conditions $U(0,t) = \sin(t+1)$, $U(1,t) = 2\sin(t+1)$, for t>0. The exact solution of this problem is $U(x,t) = (x^2+1)\sin(t+1)$.

Examples 2 : [4]

Let us consider the following space-fractional initial boundary value problem

$$\frac{\partial U(x,t)}{\partial t} = \Gamma(1.2) x^\beta \frac{\partial^\beta U(x,t)}{\partial x^\beta} + 3x^2(2x-1)e^{-t}, \tag{12}$$

with the initial condition $U(x,0) = x^2 - x^3$, and zero Dirichlet conditions. The exact solution of this problem is $U(x,t) = x^2(1-x)e^{-t}$.

The result of numerical simulations, which were obtained from implementations of the Full-Sweep Gauss-Seidel (FSGS) and Half-Sweep Gauss-Seidel (HSGS) iterative methods for examples 1 and 2 are recorded in Tables 1 and 2 respectively.

CONCLUSION

In this paper, implementation of the Half-Sweep Gauss-Seidel iterative method for solving space-fractional diffusion equations is examined. Through numerical solution Tables 1 and 2, it clearly shows that implementations of Half-Sweep Gauss-Seidel iterative concept reduce number of iterations and computational time significantly. Overall, the numerically results show that the Half-Sweep Gauss-Seidel (HSGS) method is a better method as compared to the Full-Sweep Gauss-Seidel (FSGS) methods in terms of number of iterations and execution time. For future works, we will extend this study considering this use of weighted parameter iterative methods [17,18,19,20]

REFERENCES

[1] Mainardi, F., Fractal and Fractional Calculus Continuum Mechanics, Heidelberg: Springer-Verlag, pp.291-348, 1997.

[2] Yang, B, Chen, Y & Jiang, M., *Ensemble of Differential Equations Using Pareto Optimal for Traffic Forecasting*, Telkomnika, **11(12),** pp.7309-7315, 2013.

[3] Hesameddini, E, Riahi, M and Latifizadeh, H., *A Coupling Method of Homotopy Technique and Laplace Transform for Nonlinear Fractional Differential Equations*, International Journal of Advances in Applied Science, **1(4),** pp.159-179, 2012.

[4] Azizi, H & Loghmani, G.B., *Numerical Approximation for Space-Fractional Diffusion Equations via Chebyshev Finite Difference Method,* Journal of Fractional and Applications, **4(2),** pp.303-311, 2013.

[5] Aslefallah, M & Rostamy, D., *A Numerical Scheme for Solving Space-Fractional Equation by Finite Difference Theta-Method,* International Journal of Advances in Applied Mathematics and Mechanics, **1**(4), pp.1-9, 2014.

[6] Saadatmandi, A & Dehghan, M., *A Tau Approach for Solution of The Space-Fractional Diffusion Equation,* Journal of Computer and Mathematics with Applications, **62**, pp.1135-1142, 2011.

[7] Shen, S & Liu, F., *Error Analysis of an Explicit Finite Difference Approximation for The Space-Fractional Diffusion Equation with Insulated Ends,* Journal ANZIAM, **46**(E), pp.C871-C878, 2005.

[8] Zhang, Y., A *Finite Difference Method For Fractional Partial Differential Equation,* Applied Mathematics and Computation, **215**, pp.524-529, 2009.

[9] Abdullah, A.R., *The Four Point Explicit Decoupled Group (EDG) Method: A Fast Poisson Solver,* International Journal Computer Mathematics, **38**, pp.61-70, 1991.

[10] Sulaiman, J, Hasan, M.K & Othman, M., *The Half-Sweep Iterative Alternating Decomposition Explicit (HSIADE) Method for Diffusion Equation*, Lecturer note in computer science LNCS **3314**, pp.57-63, 2004.

[11] Aruchunan, E & Sulaiman, J., *Half-Sweep Quadrature-Difference Schemes with Iterative Method in Solving Linear Fredholm Integro-Differential Equations,* Progress in Applied Mathematics , **1**, pp.11-21, 2013.

[12] Muthuvalu, M.S & Sulaiman, J., *Half-Sweep Arithmetic Method with Composite Trapezoidal Scheme for Solving Linear fredholm Integral Equations*. Journal of Applied Mathematics and Computations, **217**, pp.5442-5448, 2011

[13] Young, D.M., *Iterative Methods for Solving Partial Difference Equations of Elliptic Type,* Trans. Amer. Math. Soc., **76**, pp.92-111, 1954.

[14] Young, D.M., *Iterative Solution of large Linear Systems*, London: Academic Press, 1971

[15] Young, D.M., *Second-Degree Iterative Methods for The Solution of Large Linear Systems*, Journal of Approximation Theory, **5**, pp37-148, 1972.

[16] Sunarto, A, Sulaiman, J & Saudi, A., *Half-Sweep Gauss-Seidel Iteration Applied to Time-Fractional Diffusion Equations*, Proceeding of International Conference on Global optimization and Its applications (IcoGoia 2014), Yogyakarta, Indonesia, pp.68-74, 2014.

[17] Sunarto, A, Sulaiman, J & Saudi, A., *Full-Sweep SOR Iterative Method to Solve Space-Fractional Diffusion Equations*, Australian Journal of Basic and Applied Science, **8 (24),** pp.153-158, 2014

[18] Sunarto, A, Sulaiman, J & Saudi, A., *Implicit Finite Difference Solution for Time-Fractional Diffusion Equations using AOR Method*, Journal of Physics: Conference Series, 495, pp.012032, 2014

[19] Sunarto, A, Sulaiman, J & Saudi, A., *SOR Method for The Implicit Finite Difference Solution of Time-Fractional Diffusion Equations*, Borneo Science Journal, **34**, pp.34-42, 2014

[20] Sunarto, A, Sulaiman, J & Saudi, A., *Solving The Time-Fractional Diffusion Equations by The Half-Sweep SOR Iterative Method*, Proceeding of International Conference of Advanced Informatics: Concept, Theory and application (ICAICTA), Bandung, Indonesia pp.272-277, 2014

TABLES 1. Comparison between number of iterations (K), the execution time (seconds) and maximum errors for the iterative methods using example at $\beta = 1.2, 1.5, 1.7$

M	Method	β = 1.2			β = 1.5			β = 1.7		
		K	Time	Max Error	K	Time	Max Error	K	Time	Max Error
128	FSGS	74	1.48	2.37e-02	251	4.95	6.20e-04	930	18.29	3.99e-02
	HSGS	38	0.49	2.24e-02	98	1.15	6.69e-04	287	3.20	4.04e-02
256	FSGS	152	11.64	2.44e-02	666	51.01	5.69e-04	3029	233.01	3.97e-02
	HSGS	74	3.39	2.37e-02	251	11.27	6.21e-04	930	134.15	3.99e-02
512	FSGS	352	99.64	2.47e-02	1780	550.52	5.36e-04	9840	755.31	3.96e-02
	HSGS	152	29.26	2.44e-02	666	122.34	5.69e-04	3029	550.83	3.97e-02
1024	FSGS	709	672.27	2.49e-02	4750	1870.68	5.13e-04	21847	5259.97	3.95e-02
	HSGS	235	238.00	2.47e-02	1780	630.22	5.36e-04	9840	2215.01	3.96e-02
2048	FSGS	1547	1227.21	2.50e-02	8320	4348.68	5.02e-04	47322	8979.18	3.93e-02
	HSGS	706	534.00	2.49e-02	4741	2541.02	5.36e-04	24248	4235.80	3.96e-02

TABLES 2. Comparison between number of iterations (K), the execution time (seconds) and maximum errors for the iterative methods using example at $\beta = 1.2, 1.5, 1.7$

M	Method	β = 1.2			β = 1.5			β = 1.7		
		K	Time	Max Error	K	Time	Max Error	K	Time	Max Error
128	FSGS	57	1.42	1.80e-01	182	4.41	5.44e-02	569	13.70	8.88e-04
	HSGS	33	0.38	5.07e-02	83	0.82	1.13e-02	233	2.27	8.88e-04
256	FSGS	117	10.95	1.84e-01	481	45.32	5.58e-02	931	174.77	4.09e-04
	HSGS	63	2.50	5.28e-02	211	8.04	1.23e-02	746	28.91	4.09e-04
512	FSGS	249	93.84	1.86e-01	1277	284.40	5.65e-02	1635	427.00	1.54e-04
	HSGS	128	20.69	5.39e-02	553	86.22	1.28e-02	1390	374.84	1.54e-04
1024	FSGS	480	313.89	1.89e-01	1923	714.51	5.69e-02	5937	948.83	1.49e-04
	HSGS	271	172.33	5.45e-02	1463	570.00	1.32e-02	4619	810.72	1.25e-04
2048	FSGS	1186	557.00	1.88e-01	6241	1259.31	5.85e-02	8482	5345.02	1.20e-04
	HSGS	578	234.87	5.48e-02	3210	598.04	1.35e-02	6833	4120.13	1.20e-04